Using Matlab in quantitative analysis of yeast growth

J. Schier, B. Kovář

Affiliation Institute of Information Theory and Automation
Academy of Sciences of the Czech Republic

Abstract

Quantitative growth analysis is used in microbiology, genetics and medical research to determine the influence of the tested substance on the growth of colonies of some organism (we focus on yeast colonies in this paper). Examples of the tested substance include antibiotics, mutagens or medical drugs.

To evaluate the experiment, time series of images of the colonies inoculated on Petri dishes are taken by digital camera in a darkroom. Then, relative area and the number of the colonies in each image have to be determined for further statistical processing.

In the current laboratory practice, these parameters have often been hand-counted in each image, which is an error-prone process of a limited precision. Also, given the time demands of this hand-counting, the number of snapshots that can reasonably be processed is rather limited. In this paper, we introduce a Matlab-based tool for semi-automated processing of the dish images.

1 Introduction

Quantitative analysis of yeast growth is used in some experiments to determine the influence of a substance, contained in the growth medium, on the growth of the yeast colonies. Examples of the tested substance include antibiotics, mutagens or medical drugs. The colonies grow on a solid medium contained in a Petri dish, stored in a cultivation box (Figure 1) under constant temperature and humidity. In order to evaluate the experiment, time series of images of the colonies on each dish are taken by digital camera in a darkroom. Then, relative area and the number of the colonies in each image have to be determined for further statistical processing.

In the current laboratory practice, these parameters have often been hand-counted in each image, which is an error-prone process of a limited precision. Also, given the time demands of this hand-counting, the number of snapshots that can reasonably be processed is rather limited.

In the paper, we introduce a Matlab-based tool (see Figure 2) for semi-automated processing of the dish images.

2 Image characteristics

For each dish, images are taken several several times during the growth of the colonies. Images are taken in a dark room, using digital camera with two light sources mounted on a general-purpose imaging mount by Kaiser Fototechnik (see Figure 3). The images are characterized by the following features:

- the colonies are light-gray on a dark-grey background.
- the rim of the dish is lighter than the background, due to reflections of the illumination lamps

Figure 1: Growth box for cultivating yeast colonies

- the colonies are roughly round-shaped, there are, however, large variations in size, morphology (the surface can be either smooth or fluffy), and the exact shape, which can be nearly circular or rather fuzzy, dependent on the age of colonies and the tested substance in the medium.

While the imaging is performed in the controlled environment of a dark room, there are still some noise factors that affect each image to some degree:

- there is some degree of variation of the dish position and size, since dishes are placed manually under camera, and there are variations of the lens zoom adjustment between experiments
- there are also variations of the dish illumination (which is determined by the exact position of the dish, variations of the growth medium surface and the tilt of the lamps).

3 Processing flow of image preprocessing

The processing flow used for preprocessing is summarized in Figure 8.

In the sequel, it will be described briefly and demonstrated on the changes of an example image.

Outer background thresholding This operation is used to eliminate the background around the dish (that is, the background on which the dish is placed).

To determine the level of the background, square samples are taken in all four corners of the image. The threshold level is determined as minimum of maxima of these samples, plus as

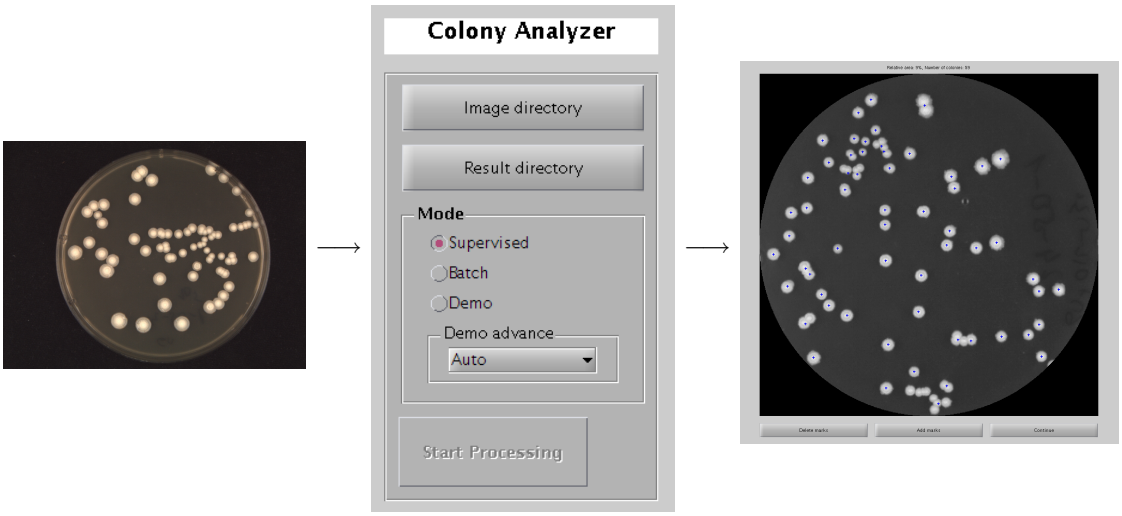


Figure 2: Matlab tool for colony counting

empirically selected guard value:

This way of calculation is used to eliminate influence of wrong values in the case of background faults (namely, misplaced black background due to operator's fault).

The original image and the image after background thresholding are shown in Figure 5.

Binary image of the dish, projections and position detection In order to determine the position of the dish, the image of the dish is converted to a binary form, and projections are calculated along horizontal and vertical axes (see Figure 6).

The coordinates of the dish are determined as the first/last point, where the value of the projection exceeds certain threshold (which is used to eliminate the influence of the background noise, shown in Figure 6d).

Out-of-image detections Once the coordinates are determined, detection of faulty images is performed, using a set of rules on the edge position and the dish radius. This allows the system to detect images missing part of the dish.

If only the dish rim is out of the image, an additional processing is used to find the position of its center and it still can be used for further processing.

Dish background thresholding Thresholding of the inner background of the dish is based on the image properties following from the arrangement of the image-taking setup and the properties of the yeast colonies: with the two lamps on the sides of the dish and with the dark background, the dish rim is brighter than the dish background, but not as bright as the colonies. Then, to cut of the background, we use the mean brightness of the image along the central horizontal line (in the approximate vertical center) along the approximate width of the rim, as the threshold (the width of the rim in pixels has been determined empirically from the images).

The process is illustrated in Figure 7, where figure (a) shows the brightness along the central horizontal line over the whole dish, figure (b) shows only the rim of the dish, and finally figure (c) the resulting binary image after thresholding.

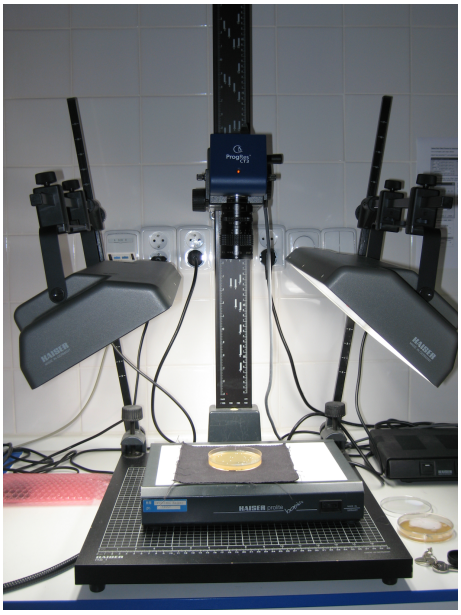


Figure 3: Imaging configuration

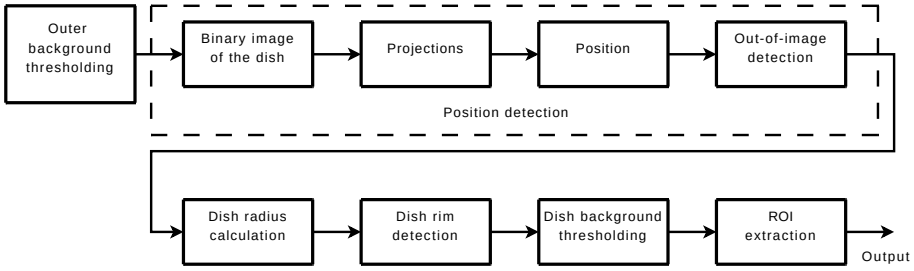
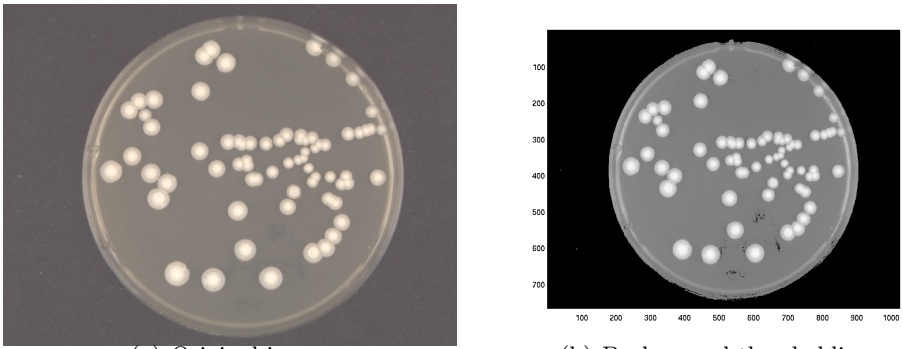


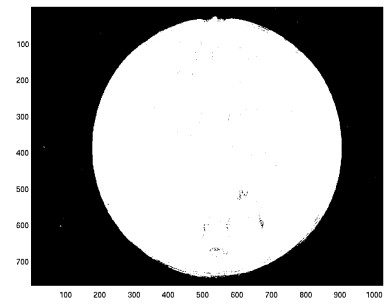
Figure 4: Processing flow of the image preprocessing



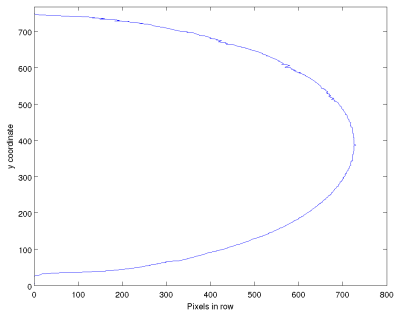
(a) Original image

(b) Background thresholding

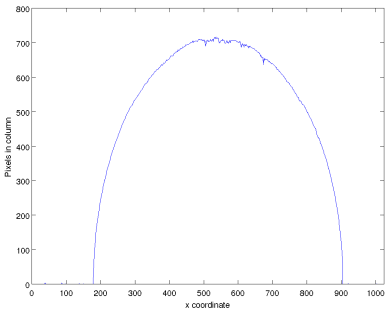
Figure 5: Original image and image after outer background thresholding



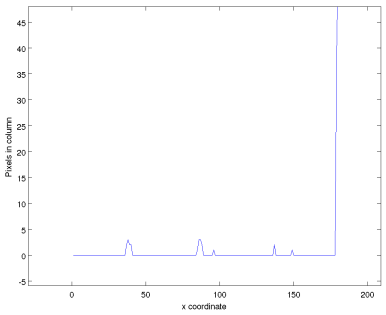
(a) Binary image



(b) Row projection

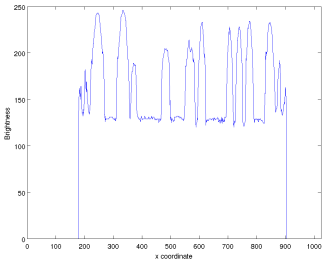


(c) Column projection

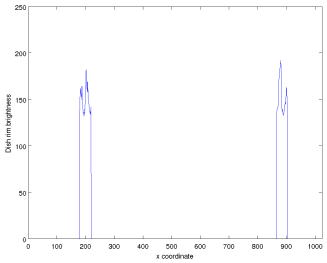


(d) Magnified part of column projection

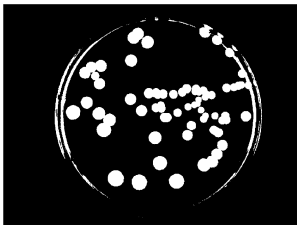
Figure 6: Projections of the dish



(a) Central line



(b) Dish rim extracted



(c) Binary image

Figure 7: Central image line and dish rim brightness, resulting binary image

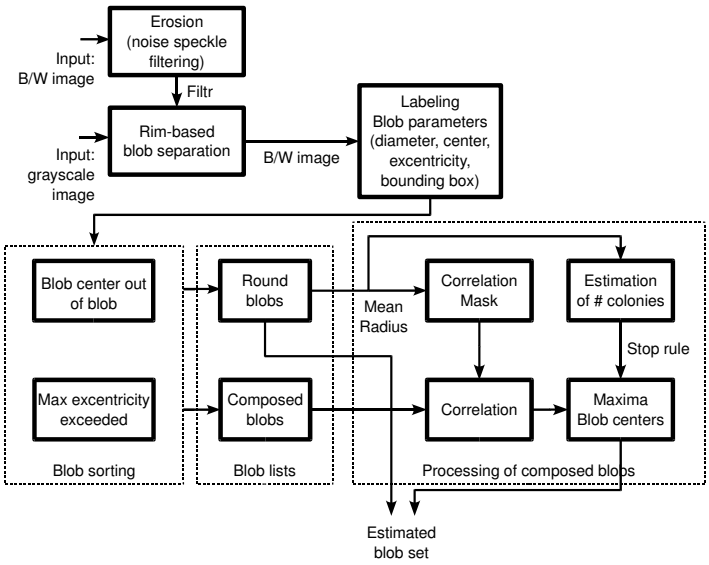


Figure 8: Processing flow of counting the colonies

4 Counting of colonies

Next step after the image preprocessing is counting of the colonies in the image. The processing flow used for counting is summarized in Figure 8.

4.1 Blob classification

The first step in counting the colonies is to sort out the isolated colonies from the touching colonies. For all colonies, equivalent diameter, center of mass, bounding box and eccentricity are computed using function `regionprops()` from the Matlab Image Processing Toolbox.

The blob centers are first checked for location out-of-blob — that is, it is tested whether the blob center lies in the position of a blob "on" pixel or in the position of a background "off" pixel (see Figure 9 for illustration). In the later case, the blob is claimed a composed one.

Next, the blobs with eccentricity exceeding a preset threshold are put to the list of composed blobs. This value of threshold is currently set to 0.8, based on test images. In future versions, this threshold should rather be a user-adjustable parameter.

4.2 Processing of composed blobs

The last step in counting is separation of the composed blobs. It is based on correlating circular mask with the contour of a composed blob. The radius of the mask is set to the mean radius of the blobs contained in the set of round blobs. The position of the individual colonies in the composed blob is then determined from the maxima of the correlation function, with the total number of colonies contained in the blob estimated from the size of the composed blob and the mean size of an individual colony. When this estimated number of colonies is reached, the search is stopped. The process is illustrated in Figure 10.

The correlation result is masked by the contours of the original blob and is further limited

Image	Colonies		Correct detection	Incorrect detection	Missed Colonies
	Total	Round			
Image 1	99	60 60.6%	79 79.8%	3 3%	20 20.2%
Image 2	66	28 42.4%	36 55%	3 4.5%	30 45%
Image 3	30	26 86.7%	27 90%	1 3.33%	3 10%
Image 4	73	48 65.8%	58 79.5%	1 1.4%	15 20.5%
Image 5	72	47 65.3%	55 76.4%	0 –	17 23.6%

Table 1: Results of colony counting

by a rectangular area of the width equal to the radius of the correlation mask, so that the maxima resulting from noise in the blob contour are suppressed as much as possible.

5 Experiments

The performance of the method has been tested on five test images. The results are summarized in Table 1.

Table 1 shows, for each test image, the following data:

- the total number of colonies (after manual correction of the counting result).
- the number of the round colonies in the image, as detected by software, also expressed as the percentage of the total number of colonies.
- Number of correctly and incorrectly detected colonies.
- Number of missed colonies.

6 Conclusions

In this paper, a Matlab software for counting the yeast colonies in a Petri dish has been described.

The performance of the method has been evaluated using five test images.

The method uses presorting of the colonies to round, isolated colonies and composed (touching) colonies, where the later are further processed to estimate the number of individual

colonies in the composed blob. Figures, showing the results of the presorting to round and composed colonies are also included.

The counting often fails on conglomerated colonies – the total miss rate is about 20%. More classification methods will be tried to improve its performance. These points, however, go beyond the scope of this paper. Finally, it should be noted that already in this phase, the tool saves time and improves accuracy of experiment evaluation in the cooperating microbiology laboratory.

For more information on the tool, do not hesitate to contact our group at `schier@utia.cas.cz`. We are interested in cooperation with possible users for further development of the tool, and for preparation of joint research projects. A version of the tool based on non-commercial implementation tools is planned, we are, however, not able to provide any release date.

The images included in the demo have been prepared in the Department of Genetics and Microbiology, Faculty of Natural Sciences, Charles University. The software has been developed in cooperation with this laboratory.

7 Acknowledgement

This paper has been supported from the project 1M0567 of the Czech Ministry of Education, Youth and Sports.

Jan Schier
Bohumír Kovář
Institute of Information Theory and Automation
Dept. of Signal Processing
Academy of Sciences of the Czech Republic
Prague 8, Czech Republic
schier@utia.cas.cz

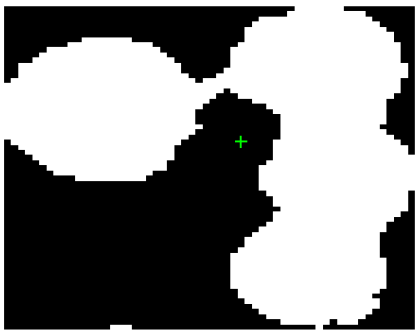


Figure 9: Illustration – center of mass out of blob

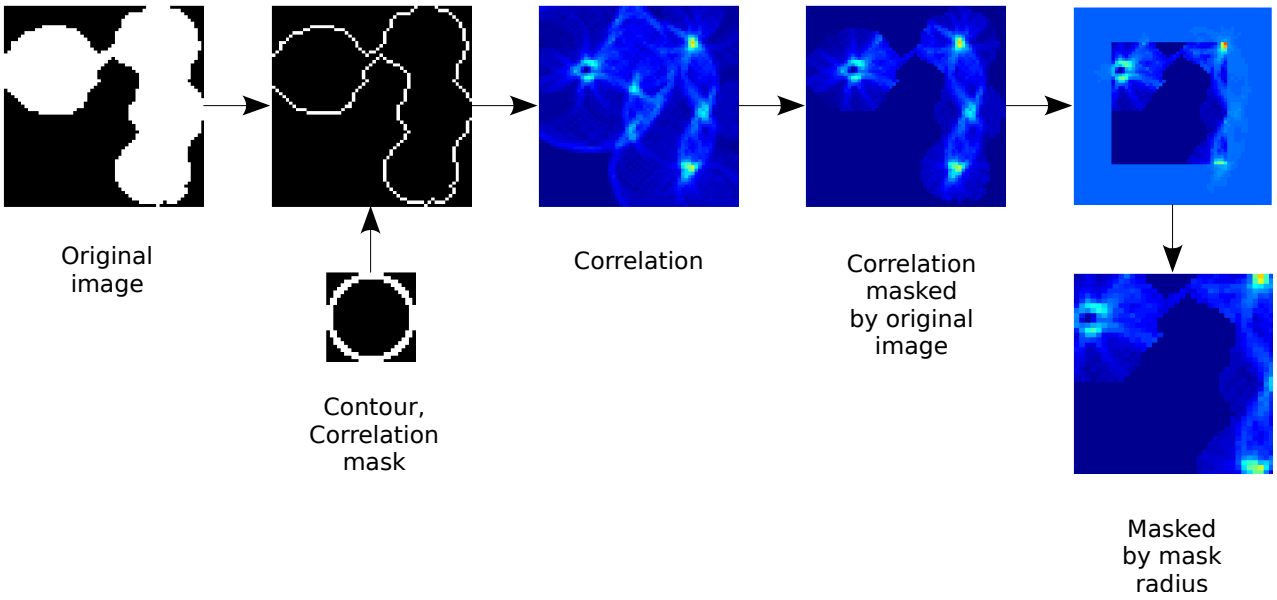


Figure 10: Blob separation flow